



TITLE:

空間の入った共同戦略(ポスター発表,基研長期研究会「複雑系」,研究会報告)

AUTHOR(S):

西村, 信一郎

---

CITATION:

西村, 信一郎. 空間の入った共同戦略(ポスター発表,基研長期研究会「複雑系」,研究会報告). 物性研究 1995, 63(6): 771-776

ISSUE DATE:

1995-03-20

URL:

<http://hdl.handle.net/2433/95503>

RIGHT:

# 空間の入った共同戦略

東大教養 西村信一郎

植物とは違って動物は動き回ることができる。魚類は海を、鳥類は空を、哺乳類は大地を。動くことが可能ということは、それ自体強力な戦略である。それも非常に高度なものだ。自力で移動するということは、敵から身を守り、餌を探せることを意味するが、それは敵と見方を見分け、餌を見つけれられることも意味する。(少なくとも「人間」にはそう見える) 何とも恐るべき能力だ。ある意味で、「自己」を認識していると言ってよいかもしれない。

このように、1匹の個体が動き回ることだけでも驚異的であるが、ここでは更に複数の個体が動き回っている状況を想定する。複数の個体が共同して狩をしたり、敵から身を守ることを考えたい。勿論、このようなことは自然界には当り前にあることで、たとえば、ライオンの集団的な狩やジャコウウシの円陣 [1] 等が挙げられる。(シマウマも円陣を作るという。ただしツノを持つジャコウウシは頭を円陣の外に向けるのに対して、シマウマはお尻を外に向け後足で防御する。) このような集団行動のアナロジーとして、できるだけ簡単なルールでモデルを作り、シミュレーションをすることにした。

## 1 モデル

### 1.1 移動のためのルール

今回のシミュレーションでは、捕食者 (predator) と被食者 (prey) が、2次元トラス上を連続的に動き回るものとして両者とも質点的に扱い、ニュートン力学に従うものとした。基本的には回りの状況を見て次に推進力をどの方向に向けるか決めればよいのであるが、その決め方が非常に難しい。敵も味方も自分の回りに複数いるわけであるし、各々が速度を持っているからである。余りよいやり方ではないが、ここでは何か平均量を見て行動を起こすことにしよう。一番簡単な方法は、自分からの敵及び味方への相対位置ベクトルの平均を取る。但し、各々自分を中心としたある円形のスコープを持っていて、その中に入っているものだけを見て平均するとしよう。これは自分の視野の中にいるだいたいの敵味方の位置を見ているというイメージである。各々は敵及び味方の平均位置ベクトルに対しどの方向に推進力を向けるかを内部にコードしてあり、それに従って行動する。以上のことを式で書くことにする。

$$\vec{a}_{P_i} = \vec{F}_{P_i} - K_{P_i} \vec{v}_{P_i} \quad (1a)$$

$$\vec{a}_{A_i} = \vec{F}_{A_i} - K_{A_i} \vec{v}_{A_i} \quad (1b)$$

$$\vec{F}_{P_i} = f[\Theta_{1P_i} \sum_{Re < |\vec{x}_{A_j} - \vec{x}_{P_i}| < R} \frac{\vec{x}_{A_j} - \vec{x}_{P_i}}{|\vec{x}_{A_j} - \vec{x}_{P_i}|} + \Theta_{2P_i} \sum_{|\vec{x}_{P_j} - \vec{x}_{P_i}| < R} \frac{\vec{x}_{P_j} - \vec{x}_{P_i}}{|\vec{x}_{P_j} - \vec{x}_{P_i}|}] \quad (1c)$$

$$\vec{F}_{A_i} = f[\Theta_{1A_i} \sum_{Re < |\vec{x}_{P_j} - \vec{x}_{A_i}| < R} \frac{\vec{x}_{P_j} - \vec{x}_{A_i}}{|\vec{x}_{P_j} - \vec{x}_{A_i}|} + \Theta_{2A_i} \sum_{|\vec{x}_{A_j} - \vec{x}_{A_i}| < R} \frac{\vec{x}_{A_j} - \vec{x}_{A_i}}{|\vec{x}_{A_j} - \vec{x}_{A_i}|}] \quad (1d)$$

$$K_{P_i} = k/E_{P_i} \quad (1e)$$

$$K_{A_i} = k/E_{A_i} \quad (1f)$$

式の説明をする。まずサフィックスの  $P_i, A_i$  はそれぞれ  $i$  番目の prey, predator を表す。第 1 式と第 2 式は運動方程式であり、 $\vec{F}$  は推進力を表し、 $K\vec{v}$  は摩擦力を表す。第 3, 第 4 式は、推進力の与え方を示す。スコープに入っている個体についてのみ相対位置を規格化して平均をとる。オペレーター  $\Theta_1, \Theta_2$  は最も重要である。これは右にあるベクトルを  $\frac{\pi}{2} (n = 0, 1, 2, 3)$  回転させるか、もしくは 0 ベクトルに変換する。 $\Theta_1, \Theta_2$  は各個体に固有のものである。第 5, 第 6 式は、その個体の「エネルギー」 $E$  が大きい程摩擦係数が小さくなることを表す。つまりエネルギーが大きい程速く走れる。「エネルギー」の説明は、§1.2 で行う。コメントしておくが、動物を質点と考えて、その全体的な運動を見ようとする研究は昔からなされており、例えば魚群 (fish school) の研究などがそれに当たる。[2, 3] 各パラメータはシミュレーションを通して次のように設定した。 $f = 0.19, E_g = 250.0, E_m = 14.6, k = 14.0$ 。

## 1.2 子孫を残すためのルール

### 1.2.1 エネルギーの導入

今回のシミュレーションでは、predator が prey を食べ、prey はそれから逃げ回ることを想定している。下手な狩をする者、下手な逃げ方をする者は淘汰されてしまう。それをモデルに入れてやることにする。各々を評価するために、指標となる「エネルギー」を導入する。predator は、餌にあり付けなければ段々エネルギーが減ってゆく。prey からエネルギーを吸い取ることで、predator は自分のエネルギーを回復させる。prey は逆に何もしなければエネルギーを増やしてゆく。これは prey が環境に存在する資源を取り入れていることに相当する。また両者は、移動するときにエネルギーを減らす。エネルギーがある閾値よりも低くなれば、その個体は死に至る。またある閾値より高くなれば、子孫を 1 匹生むことができる。(ただしその時エネルギーは等分する) 以上のことを式で表す。

$$\frac{dE_{P_i}}{dt} = E_g \exp(-\alpha N_{iP}) - kv^2 \quad (2a)$$

$$\frac{dE_{A_i}}{dt} = -E_m - kv^2 \quad (2b)$$

$$E_{P_i} \rightarrow E_{P_i} \exp(-\beta N_{iA}) \quad (2c)$$

$$E_{A_i} \rightarrow E_{A_i} + \exp(-\beta N_{iA})/N_{iA} \quad (2d)$$

第 1 式及び第 2 式は prey 及び predator のエネルギーの時間的変化を示している。第 1 式右辺第 1 項の係数  $\exp(-\beta N_{iP})$  は prey 同志の相互作用を表す。 $N_{iP}$  は、 $i$  番目の prey のスコープ(半径  $R$ )にいる他の prey の数である。つまり、密集しているほどエネルギーが増えづらい。 $-kv^2$  は個体が速く走れば走るほど、エネルギーを消費することを表している。第 3, 4 式は prey と predator のエネルギーの交換を示す。prey と predator はその距離が  $R_e (< R)$  以内である場合には、エネルギーの交換を行う。つまり prey は predator からエネルギーを吸い取られる。 $N_{iA}$  は  $i$  番目の predator の餌場(半径  $R_e$ )にいる他の predator の数である。ここで predator 同志で協力しないと捕食できないことを考えて、 $N_{iA} \geq 1$  のときのみエネルギーの交換が起こるものとしよう。即ち、近くに仲間が一匹以上いないとエネルギーを吸い取れない。注意しておきたいことは、prey は predator にエネルギーを吸い取られるだけで、即死んでしまうわけではないことである。その意味で、今回のシミュレーションは、prey-predator モデルではなく、host-parasite モデ

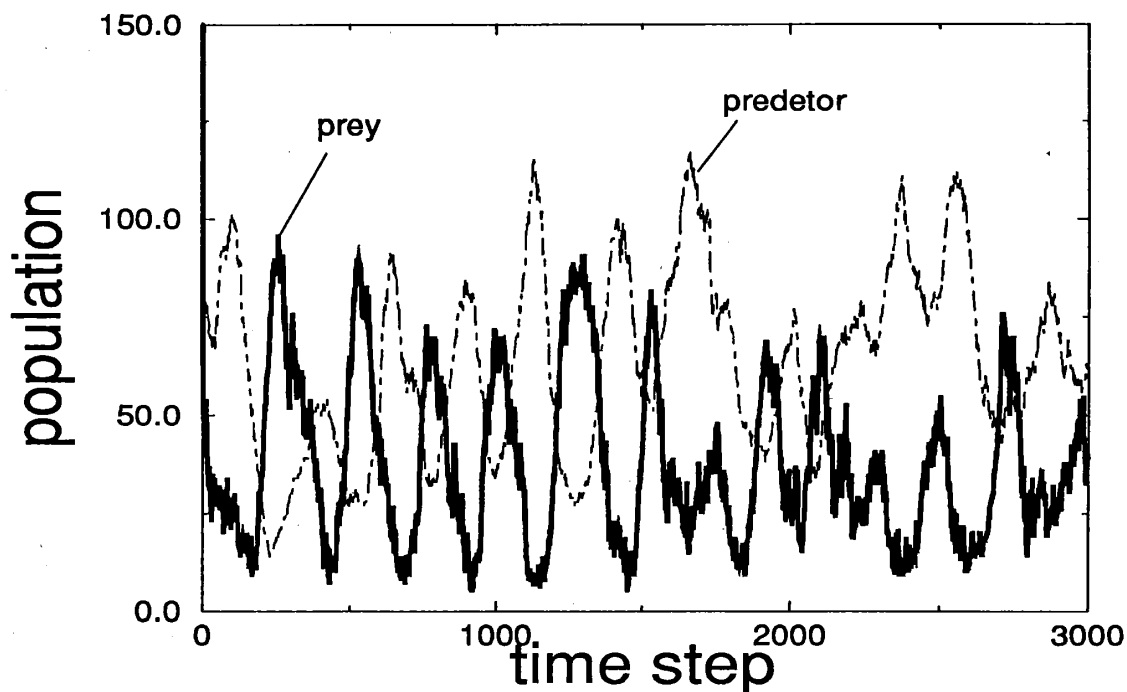


図 1: prey 及び predator の population の時間変化.  $B_P = 10.0, B_A = 65.0$

ルと言った方が良くかもしれない。いずれにせよ、重要なのは一方の存在がもう他方の存在にとって有利不利に働くことである。  $\alpha = 0.8, \beta = 0.99$  とした。

### 1.2.2 分裂, ミューテーション

§1.2.1 においても少しだけ触れたが、ここでは子孫の増やし方のルールとミューテーションについて説明する。子孫の増やし方のルールは簡単である。エネルギーがある閾値を越えたら、子どもを 1 匹自分の近くに生むことができる。(このとき速度は親と等しくする) この閾値は種全体に共通とし、prey は  $B_P$ , predator は  $B_A$  で表しておく。先述したようにエネルギーは親子とも半分となる。また、ある閾値  $D$  を下回ると、死んでしまうことにする。  $D = 5.0$  とした。

親が持つ内部コードは、そのまま子どもに引き継がれる。ただし、ある微小な確率で親とは異なるコードが引き渡される。今回のシミュレーションでは、内部コードは、 $\theta_1, \theta_2$  の二つである。これは §1.1 で説明したように、どの向きに推進力を向けるかを表すコードで、それぞれ 5 種類ある。即ち、prey predator とも 25 種類あることになる。ある微小な確率  $\mu$  で、2 つのうちどちらか一方が変異をおこす。変異のおこし方は、0 行列 or  $0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}$  回転行列 の順にサイクリック (順逆の両方) に変異する。

## 2 結果

シミュレーションの結果、ロトカ・ボルテラ型の振動相と、クラスター相の二つが出現することがわかった。その出現は  $B_P, B_A$  (得に  $B_P$ ) に大きく依存する。  $B_P$  の値がある程度低いと振動

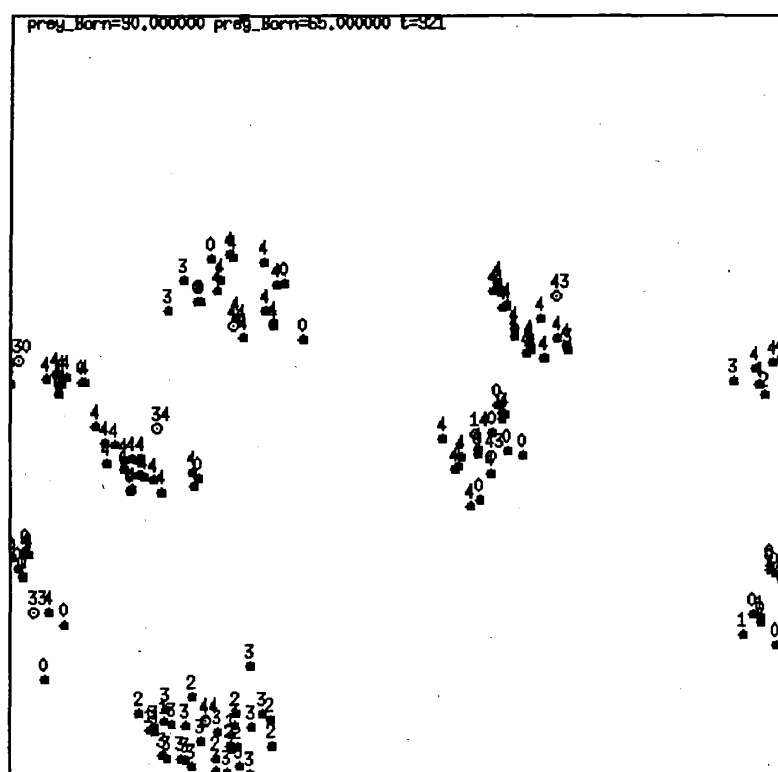


図 2: ある time step におけるスナップショット. 黒丸は predator を表す. その黒丸のクラスターの中にぽつんとある白丸が prey である.  $B_P = 90.0, B_A = 65.0$

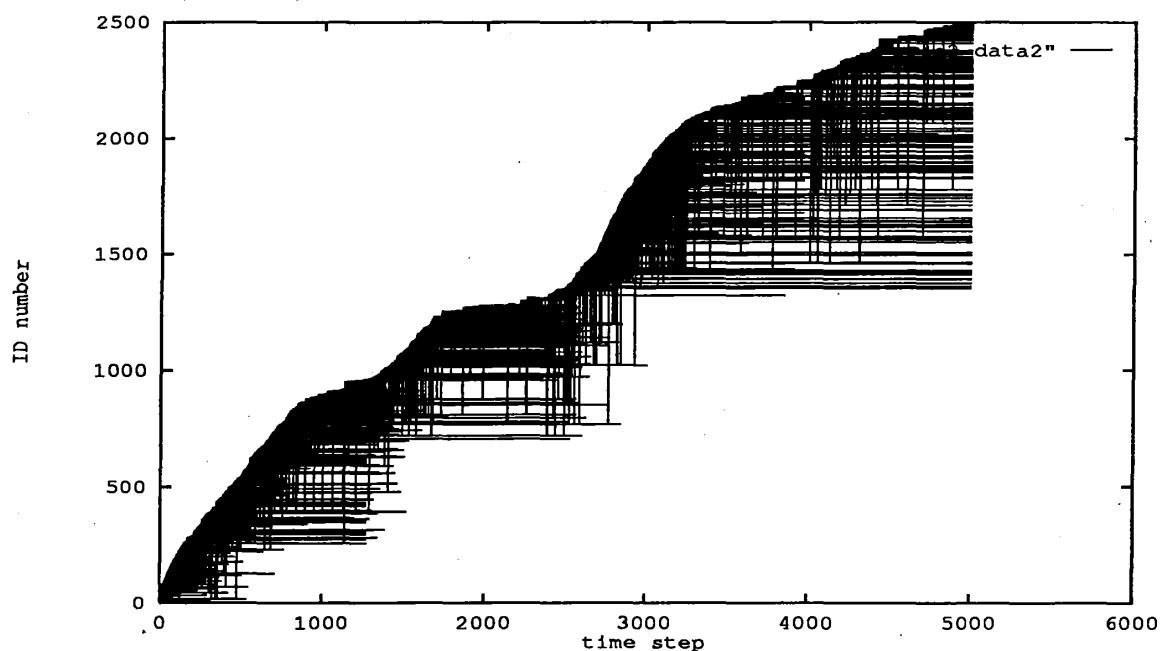


図 3: predator の系図. 横軸は time step, 縦軸は 各個体の ID number. グラフは縦線と横線からなり, 縦線はどの親から生まれたかを示し, 横線の長さは life time を表す.  $B_P = 90.0, B_A = 65.0$

相のみが見られ、高い場合には時々クラスター相が見られる。

- ロトカ・ボルテラ型の振動相

少々不規則な振動であるが、確かに prey と predator のピークは交互に出現している。(図 1) この場合、なんら特徴的な空間パターンは存在しない。ドミナントになる 内部コード は以下の通りとなる。prey は predator に対して 逆方向 ( $\Theta_1 = \pi$  回転) に進み, predator は追いかける方向 ( $\Theta_1 = 0$  回転) に進む。また味方に対して、近付こうとするものはほとんど存在しなかった。( $\Theta_2 \neq 0$  回転) これはつまり、prey は predator から逃げようとし、predator は prey を追いかけることを意味する。またそれぞれ味方に対しては、密集するのは避けようとする傾向を持つ。prey の場合、密集しているいるとそれだけ predator の気を引きやすく、(即ち、平均ベクトルがその方向を向く) 一網打尽にされてしまう。(今回のモデルでは predator は腹が一杯になることはない。) また、predator の方も、あまり仲間が近くにいない方が prey に襲いかかり易いので分散していた方が得である。

- 安定なクラスターの出現

パラメーターによっては 時々安定なクラスターが出現する。その組成は奇妙で、1 匹 もしくは 2 匹程度の prey に対し、多数の predator がその周囲を取り囲んでいる。(図 2) 少々見づらいが、白丸が prey、黒丸が predator を表す。(丸の上にある番号はそれぞれの  $\Theta_1, \Theta_2$  を表している。) 図 3 は、predator の「系図」である。(勿論パラメーターと乱数の種は 図 2 と同じである) 横軸は time step で、縦軸は predator の id number である。これは、新しい predator が生まれるたびにそれに新しい番号を割り振っていったものである。グラフは横線と縦線から構成されており、横線の長さはその個体の life time を表し、縦線はどの個体から生まれたかを示すためのものである。途中で傾斜がなだらかになっている部分はクラスターが存在していることを示している。なぜならこのクラスターは長持ちし、またほとんど新しい個体が生まれえないからである。このクラスターが現れる最大の原因は、エネルギーの交換が prey predator 間で起こっているとき、その間では力は働かないという 暗黙に作ったルールである。(式 (1c),(1d) の右辺第 1 項の和をとる範囲に下限  $R_e$  をあてたため。)

この時でも prey 同志、predator 同志では力が働くが、それを無視するコードを持っていれば、(即ち  $\Theta_2 = 0$  行列) 仲間が近くにいても止まっていられる。一度クラスターができてしまうと外から攪乱されない限り、安定して存在する。prey も predator も 動かなくなり、エネルギー収支もほぼつりあってしまうため、新しい個体はほとんど生まれて来なくなる。しかしエネルギー収支は完全にはつりあっていないため、大体においては predator の方がだんだんエネルギーを失ってゆき、やがては prey の 周囲 の predator は全滅してしまう。そうすると prey は 開放されて、クラスターは消滅する。クラスターが二つ以上ある場合には、あるクラスターの消滅は他のクラスターに影響することが多い。クラスターから開放された個体が他のクラスターのそばを通るからである。そうして、存在しているクラスターは全て消滅してしまう。以上のことが繰り返されると、図 3 のように グラフが階段状になる。

### 3 ディスカッション

今回のシミュレーションにおいて、各個体は相対位置ベクトルの平均を入力として受け取り、出力をそれに対する向きで返すように設定した。1 匹の個体が、このように極めて限られた能力しか

持っていなくとも、全体としては高度な機能を発展させうるかどうかが問題とされた。ところが、このモデルでそのような高度な機能を見出すには至らなかった。個体が持つべき能力をあまりに低く設定してしまったからなのだろうか。しかし、個体にもっと機能を持たせようとすれば、それだけモデルの複雑さが増すことになりかねない。

このモデルでは、入力集合(スコープの設定など)と出力集合(推進力など)は初めから与えてしまっている。しかもそれらを結ぶ関数(コード)も与えた。これら2つの集合をどのように選べばよいだろうか。実際の生物は入力集合も出力集合も自ら選ぶことができる。果たしてこのことをモデルに入れるのは難しいかもしれない。

今回のシミュレーションは基本的にゲームである。しかも戦略マトリックスは明示的に与えられていない。このようなゲームのルールは、それが自然なものであっても、その記述は長くなってしまう場合が多いだろう。これらの記述法というのも作る必要がある。

#### 4 なぜシミュレーションをするのか

自然を観察し、仮説を立て、それに基づいて数学的なモデルを作り、それが本物の自然に合致しているかどうか調べる、というのが自然科学本来の順序である。計算機シミュレーションをしようとも、本質的には全く変わらない。しかし、ここではもう少し計算機シミュレーションに他の意味を与えたいと思う。それは、シミュレーションを「思考の実験場」として使うことである。つまり、新しいアイデア、概念、思想の形成にシミュレーションを役立てたい。

今回のモデルも直接自然の対象物を説明するものではない。従って、モデルのルール設定にはあまり根拠がない。例えば、各個体はニュートン力学に従うとしたが、別にそうでなければならない理由は何もないのである。

上のような恣意的に作られた世界を観察者はどう見るだろうか？徹底した還元主義的な世界観で見るかもしれない。つまり要素に分解したり、機能を徹底的に調べることをするかもしれない。逆に、できるだけ全体を見ようとするかもしれない。一応モデルには prey, predator それぞれに「個体」と呼ばれる要素が存在しているが、他の観察者がこれを要素と思う必要性は何処にもない。もし要素とみなせば、各々の個体の細かい性質を調べるだろう。そうみなさなければ、何匹かの集まりを要素とするかそれとも要素などはないと考えて全体的な運動を見るかもしれない。

このように計算機シミュレーションは、「自然を観察することとは何であったか」を考え直すことに役に立つ。つまり「思考の実験場」になる。そうしているうちに何か新しいアイデア、思想が生まれてくるかもしれない。

#### 参考文献

- [1] A マニング, 動物行動学入門, 倍風館, p248
- [2] 三宮, 松田 “魚群の行動モデリング”, 計測と制御 Vol. 19, No 7, 昭和 55 年 7 月, p88-91
- [3] 三宮, 島田, 中峯 “魚群行動における自律分散機構のモデリング”, 計測自動制御学会論文集, Vol. 29, No 2, 1993, p211-219